

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re U.S. Patent Application)

Applicant: Haraguchi et al.)

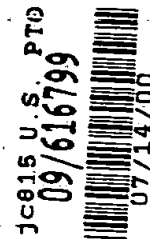
Serial No.)

Filed: July 14, 2000)

For: METHOD AND APPARATUS)
FOR ANALYZING PROGRAM)
BASED ON RULES OF...)

Art Unit:)

*I hereby certify that this paper is being deposited with
the United States Postal Service as EXPRESS MAIL in
an envelope addressed to: Assistant Commissioner for
Patents, Washington, D.C. 20231, on July 14, 2000*

Express Label No.: EL409491594Signature: *A. F.*F-CLASS.WCM
Appr. February 20, 1998Assistant Commissioner for Patents
Washington, DC 20231

Sir:

CLAIM FOR PRIORITY

Applicants claim foreign priority benefits under 35 U.S.C. § 119 on the basis
of the foreign application identified below:

Japanese Patent Application No. 11-305026

Filing Date: October 27, 1999

A certified copy of the priority document is enclosed.

Respectfully submitted,

GREER, BURNS & CRAIN, LTD.

By

Patrick G. Burns
Reg. No. 29,367

July 14, 2000
Sears Tower - Suite 8660
233 South Wacker Drive
Chicago, IL 60606
(312) 993-0080

日本国特許庁
PATENT OFFICE
JAPANESE GOVERNMENT

JCS15 U.S. PTO
09/616799



別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日
Date of Application:

1999年10月27日

出願番号
Application Number:

平成11年特許願第305026号

出願人
Applicant(s):

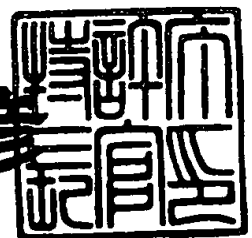
富士通株式会社

CERTIFIED COPY OF
PRIORITY DOCUMENT

2000年 3月17日

特許庁長官
Commissioner,
Patent Office

近藤隆彦



出証番号 出証特2000-3018931

【書類名】 特許願

【整理番号】 9950733

【提出日】 平成11年10月27日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 09/45

【発明の名称】 最適化促進装置

【請求項の数】 6

【発明者】

 【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

 【氏名】 原口 正寿

【発明者】

 【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

 【氏名】 林 正和

【発明者】

 【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

 【氏名】 渡辺 雄二

【特許出願人】

 【識別番号】 000005223

 【氏名又は名称】 富士通株式会社

【代理人】

 【識別番号】 100092152

 【弁理士】

 【氏名又は名称】 服部 毅巖

 【電話番号】 0426-45-6644

【手数料の表示】

 【予納台帳番号】 009874

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9705176

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 最適化促進装置

【特許請求の範囲】

【請求項 1】 構文解析されたプログラムを処理して最適化を促進させるプログラムに変換する最適化促進装置において、

各種言語に対する言語仕様情報を保持する言語仕様テーブルと、

前記言語仕様テーブルに保持された言語仕様情報を元にプログラムの言語仕様情報を解析する言語仕様情報解析手段と、

前記言語仕様情報解析手段によって解析された言語仕様解析結果情報を前記プログラムに埋め込む言語仕様解析結果情報埋め込み手段と、

を備えていることを特徴とする最適化促進装置。

【請求項 2】 前記言語仕様情報解析手段は、前記言語仕様テーブルに保持された言語仕様情報から前記プログラムの言語仕様情報を解析するのに必要な言語仕様情報解析関数を読み出しておき、前記プログラムから獲得した命令に対して前記言語仕様情報解析関数の定義からプログラム上の変数の定義または依存関係が明確な言語仕様解析結果情報を獲得することを特徴とする請求項 1 記載の最適化促進装置。

【請求項 3】 前記言語仕様解析結果情報埋め込み手段は、前記言語仕様情報解析手段が前記言語仕様テーブルから読み出した言語仕様情報解析関数を登録する関数テーブルを有することを特徴とする請求項 2 記載の最適化促進装置。

【請求項 4】 前記言語仕様解析結果情報埋め込み手段は、前記言語仕様情報解析手段が前記プログラムから獲得した命令の前に前記言語仕様解析結果情報を埋め込むようにしたことを特徴とする請求項 3 記載の最適化促進装置。

【請求項 5】 プログラムをコンパイルしてオブジェクトコードまたはソースプログラムを生成するコンパイラ装置において、

プログラムの構文解析を行うフロントエンド手段と、

各種言語に対する言語仕様情報を保持する言語仕様テーブルと、

前記言語仕様テーブルの言語仕様情報を元にプログラムの言語仕様情報を解析する言語仕様情報解析手段と、

前記言語仕様情報解析手段によって解析された言語仕様解析結果情報を前記プログラムに埋め込む言語仕様解析結果情報埋め込み手段と、

前記言語仕様解析結果情報が埋め込まれたプログラムの最適化を実施する最適化手段と、

を備えていることを特徴とするコンパイラ装置。

【請求項 6】 各種言語に対する言語仕様情報を言語仕様テーブルに保持する言語仕様情報保持手段と、前記言語仕様情報を元にプログラムの言語仕様情報を解析する言語仕様情報解析手段と、前記言語仕様情報解析手段によって解析された言語仕様解析結果情報を前記プログラムに埋め込む言語仕様解析結果情報埋め込み手段とを有する最適化促進プログラムを記録したコンピュータ読み取り可能な記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は最適化促進装置に関し、特に Fortran 90, 95 言語のように、言語仕様の定義からプログラム上の変数の値や依存関係が明確にできる言語に対して、その言語仕様を利用した解析結果情報を最適化部が利用できる形式に変換し、各最適化機能を促進させるようにした最適化促進装置に関する。

【0002】

通常、コンパイラ装置では、フロントエンド部にてソースプログラムの構文解析などを行い、最適化部でプログラムの最適化を実施し、出力装置で目的プログラムを生成する。ここで、プログラムがたとえば配列記述を含んでいる場合、そのような配列記述はループとして認識される。このとき、目的プログラムにおけるループの部分は実行時間の比重が高いため、最適化部が配列記述に関係する部分のコードを最適な状態に制御して、実行性能を改善することが行なわれている。

【0003】

【従来技術】

従来より、プログラム言語によってその表記方法が異なるが、言語仕様の定義

からプログラム上の変数の値や依存関係を明確にすることができる言語がある。その言語の例として、Fortran 90, 95 言語がある。C 言語や Fortran 77 言語などでは、配列アクセスを要素単位でしか記述できないのに対して、Fortran 90, 95 言語は、図 8 の (A) に示したような配列のスクラ表現を (B) に示したようなベクトル表現で表記することができる。ここで、言語仕様の定義からプログラム上の変数の定義や依存関係が明確にできるというのは、図 8 の (B) において、4 行目の記述から「 $L=100$ 」が成立することを意味する。

【0004】

従来の最適化技術の一つとして、ループ融合という最適化は著名な技術であり、その際、利用する配列の依存関係解析も高度な技術が存在している。ここで、ループ融合とは、所定の条件を満足する複数のループによって実行される処理を一つのループによる処理にまとめ、これによって、ループを繰り返し実行するために必要な回転制御のオーバーヘッドをループ数の減少によって削減する処理である。その依存関係解析技術を使ってループ融合の最適化を促進させる方法が既に提案されている（特開平 9-62514 号公報）。

【0005】

この方法によれば、Fortran 90, 95 のようなベクトル表現ができる言語に対して、配列の添字解析機能を用いてループ融合を促進し、最適な融合ループを作成している。たとえば図 9 において、上方に記述した 3 行が配列のベクトル表現を示しており、これに対して配列の添字解析を実行する。言語仕様上、代入文の右辺と左辺のサイズは同じであることから、「 $L=5=M-1=N-2$ 」という配列の添字解析結果情報が得られる。この配列の添字解析結果情報、すなわち「 $L=5$ 」という情報は、do 文において、ループの回数が 5 回というループ融合促進とそれによる最適なループ形状の出力とに利用されていることが分かる。

【0006】

【発明が解決しようとする課題】

このように、従来では、配列の添字解析結果情報は、ループ融合という最適化

の一つに適用することによって、最適化の促進に有効に働いたが、しかし、この配列の添字解析結果情報は他の最適化において利用されることなく捨てられていたという問題点があった。

【0 0 0 7】

本発明はこのような点に鑑みてなされたものであり、言語仕様の定義が示唆する情報を利用してさらに他の最適化が促進される最適化促進装置を提供することを目的とする。

【0 0 0 8】

【課題を解決するための手段】

図 1 は上記目的を達成する本発明の原理図である。図 1 は本発明による最適化促進装置を適用したコンパイラ装置を示している。コンパイラ装置は、プログラムソースを入力する入力部 1 と、入力されたプログラムソースの構文解析などを行うフロントエンド部 2 と、プログラムの最適化を実施する最適化部 3 と、プログラムまたはオブジェクトコードを出力する出力部 4 とから構成され、フロントエンド部 2 と最適化部 3 との間に本発明による最適化促進装置 5 が挿入されている。最適化促進装置 5 は、言語仕様情報解析部 6 と、言語仕様テーブル 7 と、言語仕様解析結果情報埋め込み部 8 とを備え、その言語仕様解析結果情報埋め込み部 8 には、関数テーブル 9 を有している。

【0 0 0 9】

この最適化促進装置 5 において、まず、言語仕様テーブル 7 には、各種言語に対する仕様の情報が保持されている。ここで、コンパイラ装置の入力部 1 にプログラムソースが投入され、フロントエンド部 2 でそのプログラムソースの構文解析などが行われるとする。最適化促進装置 5 では、言語仕様情報解析部 6 が言語仕様テーブル 7 を参照して言語仕様情報をチェックする関数を決め、そのチェック関数を言語仕様解析結果情報埋め込み部 8 の関数テーブル 9 に登録する。次に、プログラムの命令を獲得すると、言語仕様解析結果情報埋め込み部 8 は、関数テーブル 9 に登録された各チェック関数に対して言語仕様の定義からプログラムの変数の定義や依存関係が明確にできるかどうかをチェックし、言語仕様解析結果情報を獲得する。そして、得られた言語仕様解析結果情報のコードを獲得した

命令の前に埋め込む。このようにして言語仕様解析結果情報が埋め込まれたプログラムは、最適化部 3 に渡されて最適化が実施され、出力部 4 よりプログラムまたはオブジェクトコードが出力される。

【0010】

このように、最適化促進装置 5 により、言語仕様の定義からプログラムソース上の変数の定義や依存関係が明確にできる言語に対して、その言語仕様解析結果情報を最適化部 3 が利用できるコードに変換し、プログラムソースに埋め込むことにより、最適化部 3 では、変数が定数として受け取ることによる最適化の促進が可能となる。

【0011】

また、本発明によれば、各種言語に対する言語仕様情報を言語仕様テーブルに保持する言語仕様情報保持手段と、前記言語仕様情報を元にプログラムの言語仕様情報を解析する言語仕様情報解析手段と、前記言語仕様情報解析手段によって解析された言語仕様解析結果情報を前記プログラムに埋め込む言語仕様解析結果情報埋め込み手段とを有する最適化促進プログラムを記録したコンピュータ読み取り可能な記録媒体が提供される。

【0012】

この媒体に記録された最適化促進プログラムをコンピュータに実行させることにより、言語仕様情報保持手段と、言語仕様情報解析手段と、言語仕様解析結果情報埋め込み手段と、の各機能がコンピュータによって実現できる。

【0013】

【発明の実施の形態】

まず、本発明の概略について図面を参照して説明する。

図 1 は本発明による最適化促進装置の構成例を示す図である。図 1 は全体としては本発明による最適化促進装置を適用したコンパイラ装置を示している。コンパイラ装置は、基本的には、プログラムソースを入力する入力部 1 と、入力されたプログラムソースに対して構文解析などを実施するフロントエンド部 2 と、プログラムの最適化を実施する最適化部 3 と、プログラムまたはオブジェクトコードを出力する出力部 4 とから構成されており、これに本発明による最適化促進装

置 5 が実装されている。

【0014】

最適化促進装置 5 はフロントエンド部 2 と最適化部 3 との間に挿入され、言語仕様情報解析部 6 と、言語仕様テーブル 7 と、言語仕様解析結果情報埋め込み部 8 とを備えている。その言語仕様解析結果情報埋め込み部 8 には、関数テーブル 9 を有している。また、この最適化促進装置 5 における言語仕様テーブル 7 には、各種言語に対する言語仕様の情報があらかじめ保持されている。

【0015】

以上の構成において、コンパイラ装置の入力部 1 にプログラムソースが投入されると、フロントエンド部 2 ではそのプログラムソースに対して構文解析などが行われる。次に、最適化促進装置 5 では、言語仕様情報解析部 6 が言語仕様テーブル 7 を参照して言語仕様情報を解析する言語仕様情報解析関数（以下チェック関数という）を決め、そのチェック関数を言語仕様解析結果情報埋め込み部 8 の関数テーブル 9 に登録する。次に、言語仕様情報解析部 6 がプログラムの命令を獲得すると、言語仕様解析結果情報埋め込み部 8 は、関数テーブル 9 に登録された各チェック関数に対して言語仕様の定義からプログラムの変数の定義や依存関係が明確にできるかどうかをチェックし、言語仕様解析結果情報を獲得する。そして、獲得された言語仕様解析結果情報のコードを獲得した命令の前に埋め込む。このようにして言語仕様解析結果情報が埋め込まれたプログラムは、最適化部 3 に渡されて最適化が実施され、出力部 4 よりプログラムまたはオブジェクトコードが出力される。

【0016】

このように、最適化促進装置 5 は、言語仕様テーブル 7 に保持された言語仕様の定義からプログラムソース上の変数の定義や依存関係が明確な言語仕様解析結果情報を最適化部 3 が利用できるコードに変換し、プログラムソースに埋め込むようにした。これにより、最適化部 3 では、変数として渡されていた言語仕様情報が定数として渡されるようになる。この結果、最適化部 3 で実施される最適化は、たとえばループ融合のような特定の最適化が促進されるのではなく、複数の最適化が促進される可能性が出てくる。

【0017】

次に、本発明の実施の形態をさらに詳細に説明する。まず、言語仕様テーブル7が保持する言語仕様情報の例について説明する。

図2は言語仕様テーブルの例を示す図であって、(A)は言語仕様情報の第1のテーブルの例を示し、(B)は言語仕様情報の第2のテーブルの例を示している。図2の(A)に示した言語仕様情報の第1のテーブルによれば、言語仕様情報解析関数(チェック関数)として、たとえば「配列のベクトル表記」が各言語に対して可能かどうかを定義している。また、(B)に示した言語仕様情報の第2のテーブルによれば、言語仕様情報解析関数(チェック関数)として、たとえばNo. 1に「配列代入文において、参照側の結果サイズと定義側のサイズは等しい」ということが各言語に対して当てはまるかどうかを定義している。

【0018】

図3はプログラムソースの一例を示す図である。このプログラムソースの4行目および5行目の命令に注目すると、配列代入文の記述がある。配列代入文は、言語仕様情報解析部6による解析の結果、Fortran90およびFortran95言語では、右辺と左辺のサイズが等しいことから、Lは5に等しく、Nは6に等しいことが分かる。本発明では、これら言語仕様解析結果情報である「L=5」および「N=6」を最適化部3に伝えることで、最適化の促進を図ることができる。この例では、プログラムソースの6行目に「do i=1, L」の記述がなされているが、その記述の中の変数Lは5であることを最適化部3に伝えることになる。その伝え方は、言語仕様解析結果情報埋め込み部8により、プログラムに言語仕様解析結果情報である「L=5」および「N=6」を埋め込むことによって行う。

【0019】

図4は言語仕様解析結果情報を埋め込んだプログラムソースを示す図である。この図4によれば、図3に示すプログラムに対して、LおよびNを含む命令の前に、「L=5」、「N=6」という言語仕様解析結果情報がそれぞれ埋め込まれている。この例では、実際に最適化部3に伝えられて有効になるのは、「L=5」の言語仕様解析結果情報だけである。

【0020】

ここで、具体的な効果を明示するために、「 $L=5$ 」という情報を最適化部3に伝えることで、促進される最適化について説明する。

図5はループの全展開最適化を適用した場合の例を示す図である。この図5のプログラムは、「 $L=5$ 」という情報をプログラムソースの6行目のdoループの終値として適用することにより、ループアンローリング最適化が促進された関数本体部分を示している。図5によれば、ループアンローリング最適化が実施されて、ループは全展開され、プログラム上にループはなくなっている。

【0021】

次に、図5のループアンローリング最適化を実施したプログラムに対して、さらに並列度を高める最適化を適用した例を示す。

図6は並列度を向上させる最適化を適用した場合の例を示す図である。図5のプログラムに対して演算順序を変更することにより、並列度を高める最適化を適用すると、図6のプログラムのように最適化される。すなわち、図5のプログラムでは、 k の演算が連なっているが、これらは、それぞれ前の演算が終了しないと、後ろの演算を実行することはできない。しかし、図6のプログラムでは、「 $k = k + a(1)$ 」、「 $k1 = a(2) + a(3)$ 」および「 $k2 = a(4) + a(5)$ 」という記述に注目してみると、たとえば「 $k1 = a(2) + a(3)$ 」という演算は、その上の「 $k = k + a(1)$ 」という演算の結果を使わないので、上の演算と関係なく演算を実行することができる。その下の「 $k2 = a(4) + a(5)$ 」という演算も上の演算に依存せずに演算を実行することができる。したがって、この三つの演算は、複数の足し算を一回で発行できるようなマシンにおいて、複数の足し算を並行して行うことができるような最適化が実施されたことになる。ただし、配列 a 、変数 k が浮動少数点数である場合、この最適化は精度誤差が生じる可能性がある。

【0022】

このように、プログラムから明確となる言語仕様解析結果情報である「 $L=5$ 」をプログラムに埋め込むだけで、後に記述されたdoループの終値に定数伝搬することができ、従来技術の最適化が促進されるのである。

【 0 0 2 3 】

次に、最適化促進装置の処理の流れについて図 3 に示したプログラムソースが
入力された場合を例に説明する。この場合、処理のターゲットであるプログラム
は、Fortran 90 言語である。

【 0 0 2 4 】

図 7 は最適化促進装置のアルゴリズムを示すフローチャートである。まず、言
語仕様情報解析部 6 は、言語仕様テーブル 7 を元に、ターゲットである言語仕様
を利用して情報解析を行うチェック関数（言語仕様情報解析関数）を決定し、そ
れを言語仕様解析結果情報埋め込み部 8 の関数テーブル 9 へ登録を行う（ステッ
プ S 1）。すなわち、図 2 に示した二つのテーブルが参照され、そこから、第 2
のテーブルにおける No. 1 および No. 2 のチェック関数が関数テーブル 9 に
登録される。次に、言語仕様情報解析部 6 は、プログラムから命令を獲得し（ス
テップ S 2）、獲得した命令が最後の命令かどうかを判断する（ステップ S 3）
。最後の命令であれば、この処理は終了し、最後の命令でなければ、カウンタ値
i をその初期値である「1」に設定する（ステップ S 4）。次に、i がチェック
関数テーブルサイズ以下かどうか判断される（ステップ S 5）。ここでは、チ
ェック関数テーブルサイズは、関数テーブル 9 には No. 1 および No. 2 のチ
ェック関数が登録されているので、「2」である。i がチェック関数テーブルサ
イズを越えていれば、ステップ S 2 に戻り、i がチェック関数テーブルサイズ
以下であれば、言語仕様情報解析部 6 は関数テーブル 9 から i 番目のチェック関数
を呼び出し（ステップ S 6）、プログラム内の言語仕様情報の解析を行い、言語
仕様解析結果情報を獲得する（ステップ S 7）。すなわち、プログラムソースの
4 行目の命令に対して「L = 5」、5 行目の命令に対して「N = 6」という言語
仕様解析結果情報を獲得する。次に、言語仕様解析結果情報埋め込み部 8 は、獲
得した言語仕様解析結果情報をプログラムソースに埋め込む（ステップ S 8）。
この埋め込みは、獲得した言語仕様解析結果情報を獲得した命令の前の行に対
して行う。そして、カウンタ値 i をインクリメントし（ステップ S 9）、ステッ
プ S 5 に戻る。

【 0 0 2 5 】

このようにして、登録されたチェック関数すべてについて、言語仕様情報の解析および言語仕様解析結果情報の獲得を行い、プログラムソースへの埋め込みを行うことで、図4に示したプログラムソースが得られる。この言語仕様解析結果情報が埋め込まれたプログラムが最適化部3への入力となり、この結果、図5および図6に示したような最適化が可能になる。

【0026】

なお、以上の実施の形態では、最適化促進装置5への入力は、プログラムソースの場合を示したが、これに限らず、コンパイラ中間言語であってもよい。

また、以上の実施の形態では、言語仕様情報の解析結果を表す情報を最適化部3へ伝える方法として、言語仕様情報解析結果情報をプログラムソースに埋め込むやり方を例示したが、得られた言語仕様情報解析結果情報をテーブルに保持しておいて、その情報を最適化部3に渡すという方法もある。ただし、この言語仕様解析結果情報テーブルを利用する方法は、ある変数の値がいくつでその有効範囲がどこからどこまでであるという情報を求めるための手間のかかる解析が必要であることと、言語仕様解析結果情報埋め込み機能では、既存最適化にプラグインする形で適用できるが、言語仕様解析結果情報テーブルを利用するやり方は、最適化自体、そのテーブルの検索をし、ある命令の時点でそれが参照する変数の値が分かるかどうかをチェックする処理が必要となる。

【0027】

さらに、上記の最適化促進装置5の処理内容は、コンピュータで読み取り可能な記録媒体に記録されたプログラムに記述しておくことができる。このプログラムをコンピュータで実行することにより、上記処理がコンピュータで実現できる。コンピュータで読み取り可能な記録媒体としては、磁気記録装置や半導体メモリなどがある。市場に流通させる場合には、CD-ROM (Compact Disk Read Only Memory) やフロッピーディスクなどの可搬型記録媒体にプログラムを格納して流通させたり、ネットワークを介して接続されたコンピュータの記憶装置に格納しておき、ネットワークを通じて他のコンピュータに転送することもできる。コンピュータで実行する際には、コンピュータ内のハードディスク装置などにプログラムを格納しておき、メインメモリにロードして実行する。

【0 0 2 8】

【発明の効果】

以上説明したように本発明では、言語仕様の定義からプログラム内の変数の値や依存関係が明確にできる言語に対して、その言語仕様を利用した解析結果情報を最適化部で利用できる形式に変換するように構成した。このため、従来の技術では不可能であった様々な最適化を促進させることができるようになる。

【0 0 2 9】

また、促進される最適化は特定の最適化が促進されるわけではなく、複数の最適化が促進される可能性があり、その効果は非常に大きい。

【図面の簡単な説明】

【図 1】

本発明による最適化促進装置の構成例を示す図である。

【図 2】

言語仕様テーブルの例を示す図であって、(A)は言語仕様情報の第1のテーブルの例を示し、(B)は言語仕様情報の第2のテーブルの例を示している。

【図 3】

プログラムソースの一例を示す図である。

【図 4】

言語仕様解析結果情報を埋め込んだプログラムソースを示す図である。

【図 5】

ループの全展開最適化を適用した場合の例を示す図である。

【図 6】

並列度を向上させる最適化を適用した場合の例を示す図である。

【図 7】

最適化促進装置のアルゴリズムを示すフローチャートである。

【図 8】

配列の記述例を示す図であって、(A)は配列のスカラー表現を示し、(B)は配列のベクトル表現を示している。

【図 9】

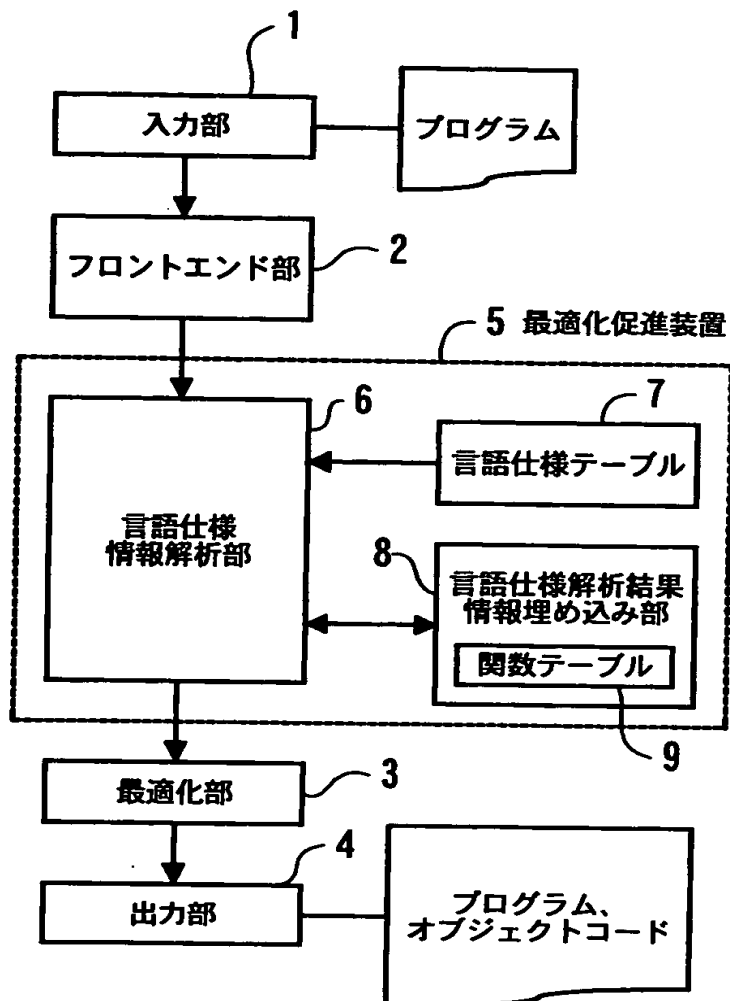
配列の添字解析機能を用いたループ融合促進および最適なループ形状の決定の例を示す図である。

【符号の説明】

- 1 入力部
- 2 フロントエンド部
- 3 最適化部
- 4 出力部
- 5 最適化促進装置
- 6 言語仕様情報解析部
- 7 言語仕様テーブル
- 8 言語仕様解析結果情報埋め込み部
- 9 関数テーブル

【書類名】 図面

【図 1】



【図 2】

(A)

言語	Fortran77	Fortran90	Fortran95	C	Java	□□	△△
配列のベクトル表記	×	○	○	×	×	○	×
:	:	:	:	:	:	:	:

(B)

No.	言語	Fortran90	Fortran95	□□
1	配列代入文において、参照側の結果 サイズと定義側のサイズは等しい	○	○	×
2	where 文のマスクサイズと where 内の命令の定義側サイズは等しい	○	○	×
:	:	:	:	:

【図 3】

プログラムソースの例

```

1 subroutine sub(a,b,k,L,N)
2 integer(kind=4), dimension(1:6) :: a,b
3 integer(kind=4) :: L,N
4 a(2:L) = b(2:5)
5 b(3:N) = a(1:4)
6 do i=1,L
7   k = k + a(i)
8 enddo
9 end subroutine

```

【図 4】

言語仕様解析結果情報の埋め込み

```

subroutine sub(a,b,k,L,N)
integer(kind=4), dimension(1:6) :: a,b
integer(kind=4) :: L,N
L=5
a(2:L) = b(2:5)
N=6
b(3:N) = a(1:4)
do i=1,L
  k = k + a(i)
enddo

```

【図 5】

ループの全展開最適化の適用結果

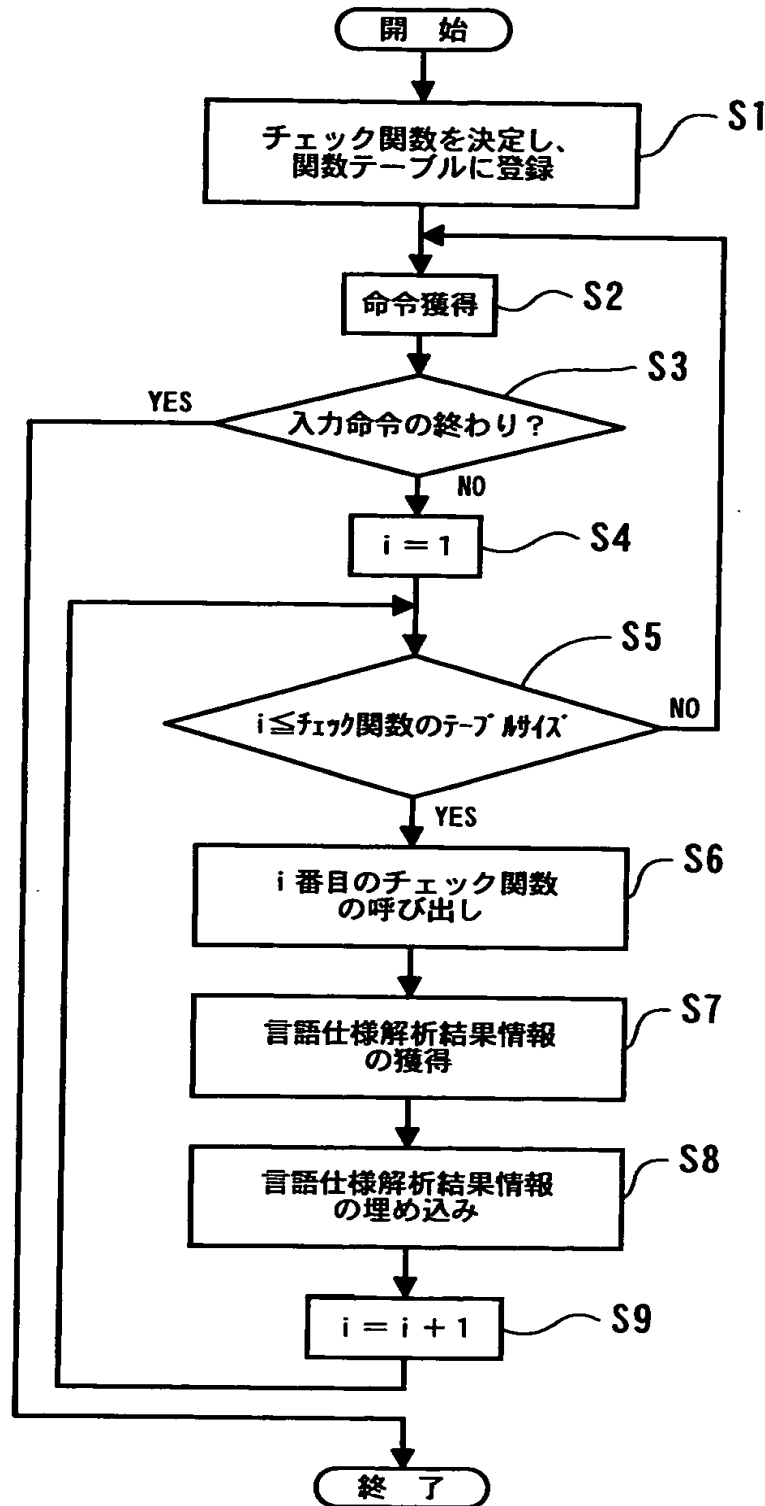
$a(2) = b(2)$
 $a(3) = b(3)$
 $a(4) = b(4)$
 $a(5) = b(5)$
 $b(3) = a(1)$
 $b(4) = a(2)$
 $b(5) = a(3)$
 $b(6) = a(4)$
 $k = k + a(1)$
 $k = k + a(2)$
 $k = k + a(3)$
 $k = k + a(4)$
 $k = k + a(5)$

【図 6】

並列度を向上させる最適化の適用結果

```
a(2) = b(2)
a(3) = b(3)
a(4) = b(4)
a(5) = b(5)
b(3) = a(1)
b(4) = a(2)
b(5) = a(3)
b(6) = a(4)
k = k + a(1)
k1 = a(2) + a(3)
k2 = a(4) + a(5)
k = k + k1
k = k + k2
```

【図 7】



【図 8】

(A) 配列のスカラー表現

```
1 subroutine sub(r,q)
2 real r(100),q(100)
3 do i=1,100
4 r(i)=q(i)
5 enddo
6 end subroutine
```

(B) 配列のベクトル表現

```
1 subroutine sub(r,q,L)
2 real r(100),q(100)
3 integer L
4 r(1:L)=q(1:100)
5 end subroutine
```

【図 9】

```
a(1:L)=b(2:6)
b(2:M)=c(3:L+2)
c(3:N)=a(1:M-1)
```

↓ L=5=M-1=N-2 (配列の添字解析結果)

```
do i=1, 5 ! 最適なループ形状の出力
  a(i)=b(i+1)
  b(i+1)=c(i+2)
  c(i+2)=a(i)
enddo
```


【書類名】 要約書

【要約】

【課題】 最適化促進装置に関し、言語仕様の定義が示唆する情報を利用して最適化が促進されるようにすることを目的とする。

【解決手段】 最適化促進装置 5 はコンパイラ装置におけるフロントエンド部 2 と最適化部 3 との間に挿入される。言語仕様テーブル 7 は、各種言語に対する言語仕様の情報を保持する。言語仕様情報解析部 6 は、言語仕様情報解析関数を言語仕様テーブル 7 から得て、関数テーブル 9 に登録する。次に、フロントエンド部 2 から渡されたプログラムから命令を獲得し、その命令に対して、登録された言語仕様情報解析関数を解析して言語仕様解析結果情報を取得し、それを言語仕様解析結果情報埋め込み部 8 がプログラムに埋め込む。言語仕様解析結果情報が埋め込まれたプログラムは、言語仕様情報解析部 6 から最適化部 3 に渡されて様々な最適化が実施される。

【選択図】 図 1

出 願 人 履 歴 情 報

識別番号 〔000005223〕

1. 変更年月日	1996年 3月26日
[変更理由]	住所変更
住 所	神奈川県川崎市中原区上小田中4丁目1番1号
氏 名	富士通株式会社